

Introdução:

Slide: Computação gráfica

Fala: A computação gráfica é a área da computação destinada à geração de imagens em forma de representação de dados e informação, ou em forma de recriação do mundo real, é baseada em pixels que são pontos, e ao se juntar, fazem com que a imagem seja sintetizada visualmente em telas e monitores, esta área está sendo muito implementada em nosso cotidiano podendo ser visualizada cada vez mais sua presença.

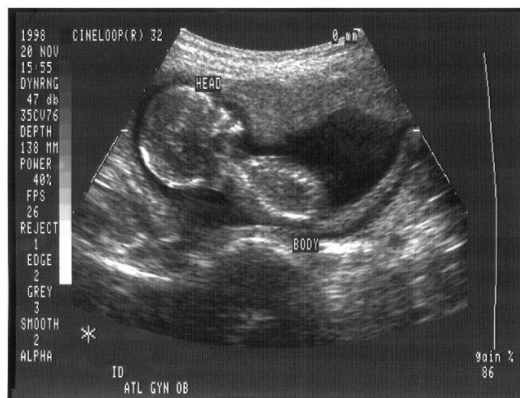
Na Computação Gráfica existem três grandes áreas:

Slide: Síntese de Imagens: busca gerar imagens de objetos e cenas próximas do real



Fala: Exemplo: cinema, jogos, arquitetura

Slide: Análise de imagens: procura obter a especificação dos componentes de uma imagem a partir de sua representação visual



Fala: Exemplo: Medicina

Fala: E a área de abordaremos em nosso trabalho

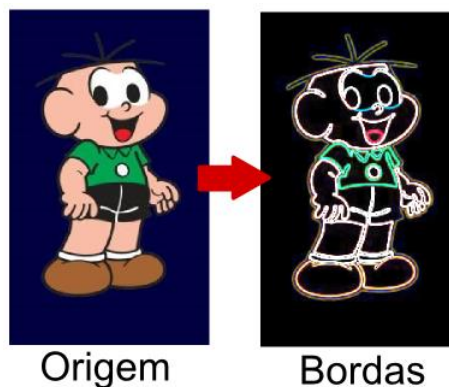
Slide: Processamento de Imagens:



Fala: Exemplo: Arte, Designer, medição e detecção de objetos,

Fala: O processamento de uma imagem tende a ser um procedimento de dados de entrada e saída, consiste na transformação de uma imagem realçando os fatores de interesse necessitados, removendo os dados indesejáveis, o que também pode ser chamado de segmentação de uma imagem, no qual em nosso trabalho será realizado pela detecção de bordas em uma imagem.

Slide: Detecção de bordas



Fala: Uma borda é caracterizada por mudança nos níveis de intensidade dos pixels onde a intensidade luminosa muda repentinamente. A detecção de borda é definida para encontrar esse tipo de variação dos pixels, e esses pixels quando próximos podem ser conectados formando uma borda ou contorno e assim definindo a região de um objeto, sua dimensão e variação.

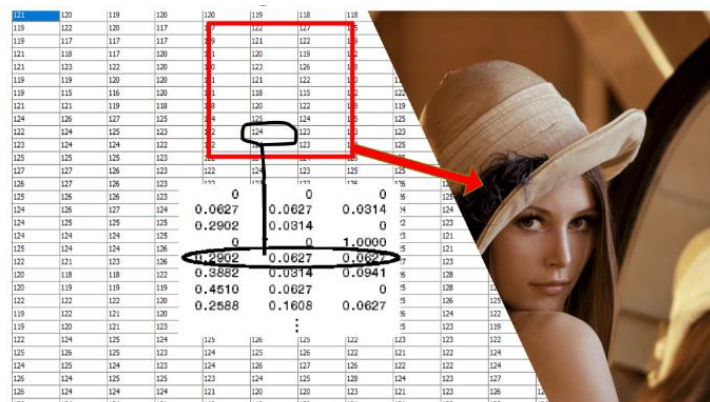
Processamento da imagem

Slide: Leitura da Imagem:

Fala:

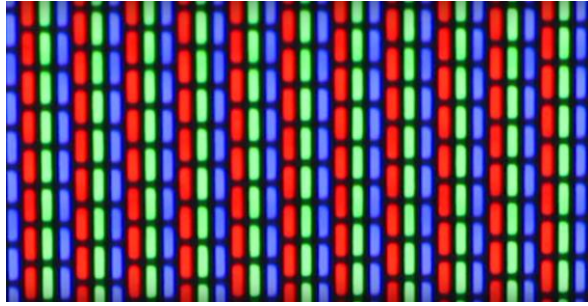
- Uma imagem é armazenada pelos os programas de forma matricial. Isso possibilita realizar operações sobre a fotografia, como soma matricial, multiplicação por escalar e etc.
- Cada elemento recebe o nome de pixel;

Slide: Imagem colorida



//Aproximar tela





Fala:

- O tipo de imagem utilizadas (8 bits) possibilita ter uma variação de 0 a 255 cores;
- Em imagens coloridas o pixel possui 3 bandas: Red, blue and Gree. Cada ponto é um resultante dessas três cores;

Slide: Imagem em tons de cinza

Fala:

- Para se trabalhar com uma imagem é necessária transforma-la em monocromática, tornando – a uma imagem em tons de cinza
- Utiliza-se uma média pondera
 - $F(x,y) = 0,299 * Fr(x,y) + 0,587 * Fg(x,y) + 0,114 * Fb(x,y)$



→ Imagem binária

- Após o processamento o resultado é uma imagem binária, tal que 1 representa borda, em tonalidade de branco, e 0 representa ausência de borda na tonalidade de preto.

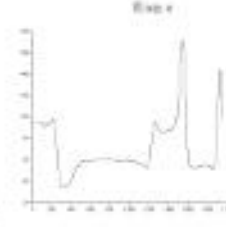
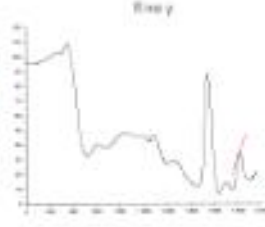
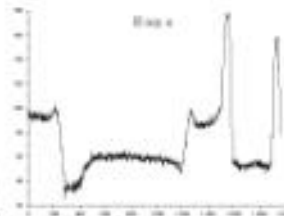
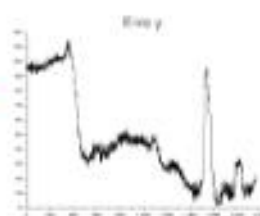


0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	1	1	0	0	0	0	0
0	1	1	0	0	0	1	1	0	0	0	0	0	0
0	0	1	1	1	0	1	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0

Método Canny:

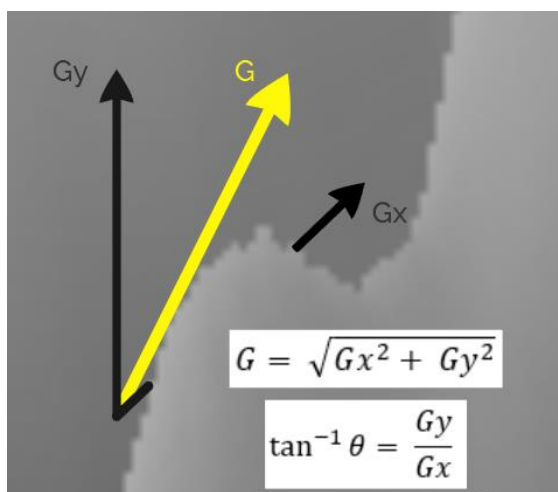
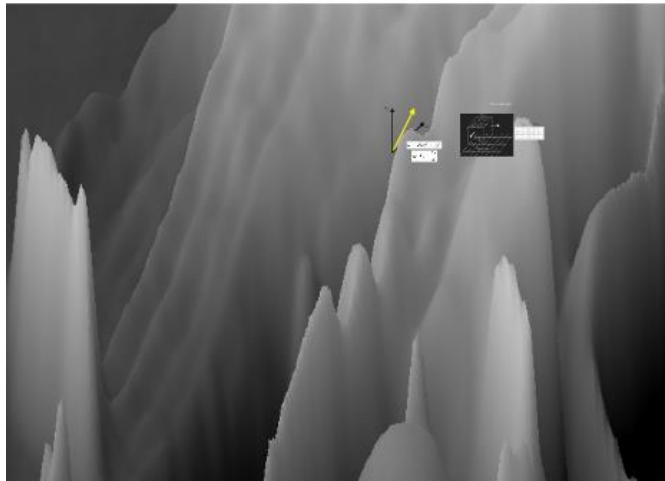
Slide: Suavização Gaussiana

Fala: A suavização Gaussiana tem como objetivo diminuir ruídos e variações bruscas nas cores da imagem. Após a suavização é possível identificar com maior precisão a posição das bordas na imagem.



Slide: Gradiente.

Fala: Uma imagem é constituída por duas dimensões o que faz necessário implementar o conceito de gradiente para identificar onde há a maior variação de cores.



Slide: convolução

Fala: Para calcular o gradiente é necessário fazer operações de convolução sobre a imagem, que nada mais é do que varrer uma imagem utilizando uma matriz de operação (kernel). O elemento a ser analisa é centrado na matriz kernel, há a multiplicação de cada elemento pelo seu correspondente, por fim faz se o somatório dos elementos. O valor desse somatório é substituído na posição do pixel na matriz.

Imagem:

100	100	50	70
100	100	50	70
100	100	50	70
100	100	50	70
100	100	50	70

G_y

-1	-2	-1
0	0	0
1	2	1

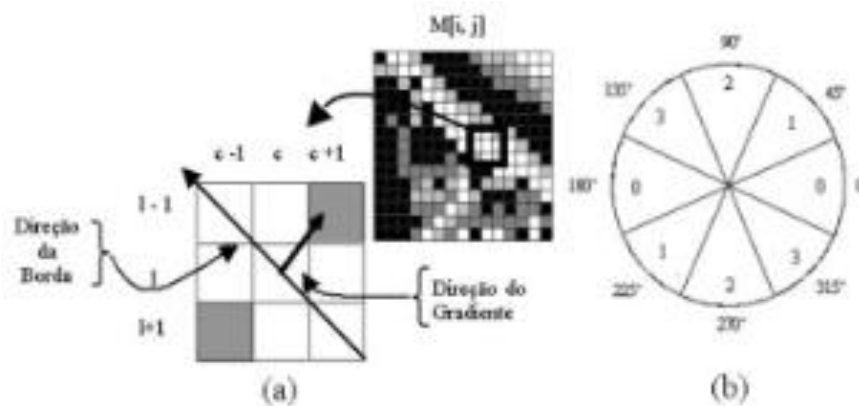
Saída

100	-200	-50
0	0	0
100	200	50

$G_y = 0$

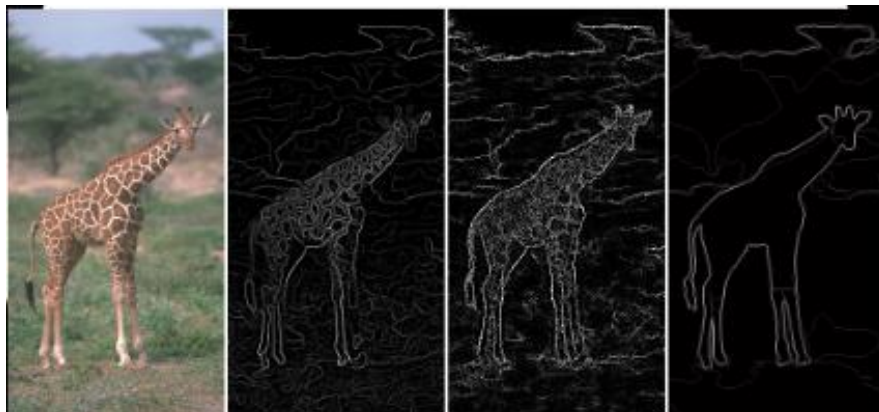
Slide: Supressão do não máximo

Fala: A supressão do não máximo pega todos os valores na direção do gradiente, zerando aqueles que não são máximos;



Slide: Limiarização por histerese

Fala: A limiarização por histerese escolhe um intervalo aceitável como sendo borda, depois disso faz uma binarização da foto, transformando-a em um imagem de bordas.



→ Resultados

Slide

Correlação entre alturas

Fala: Foram realizadas diversas experimentações para constatar a correlação entre a altura em pixels, e a altura real do objeto em milímetros. Para esse processo, foi necessário o controle de algumas variáveis, como: a distância do objeto à câmera; a rotação do objeto; e o processo de limiarização do método Canny.

OBS: As diferenças entre as tabelas serão todas explicadas.

Objeto	Tamanho (mm)	Medida em pixels	Relação Pixel/mm
A4 horizontal	210	280	0,75
A4 vertical	294	394	0,75
Celular	140	185	0,76
Cartão vertical	94	124	0,76
Cartão horizontal	66	88	0,75

Média:	0,75
Desv. Padrão:	0,005

Tabela 1: Resultado das medições realizadas à 500 milímetros de distância.

Objeto	Tamanho (mm)	Medida em pixels	Relação Pixel/mm
A4 horizontal	210	191	1,10
A4 vertical	294	269	1,09
Celular	140	126	1,11
Cartão vertical	94	86	1,09
Cartão horizontal	66	61	1,08

Média:	1,10
Desv. Padrão:	0,011

Tabela 2: Resultado das medições realizadas à 750 milímetros de distância.

Objeto	Tamanho (mm)	Medida em pixels	Relação Pixel/mm
A4 horizontal	210	142	1,48
A4 vertical	294	188	1,56
Celular	140	91	1,54
Cartão vertical	94	64	1,47
Cartão horizontal	66	45	1,47

Slide

PIXEL/MLÍMETRO

Fala

Foi necessário criar uma etapa para estabelecer uma relação pixel/milímetro inicial para ser mantida durante a medição de novos objetos. Este processo consiste na utilização de um objeto com dimensões conhecidas, e a partir da medição deste objeto dada em pixels computar a relação entre pixel e milímetro a ser mantida.

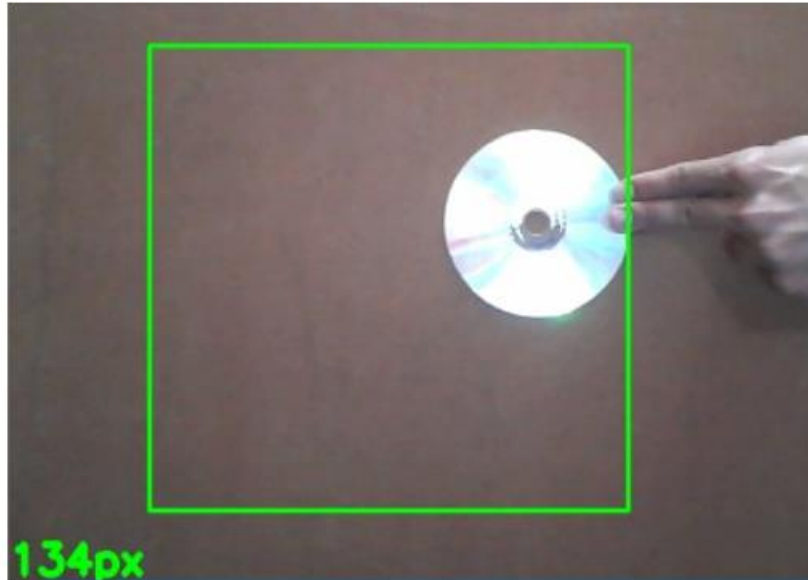


Figura 4: Computando altura em pixels para ser utilizado como referência

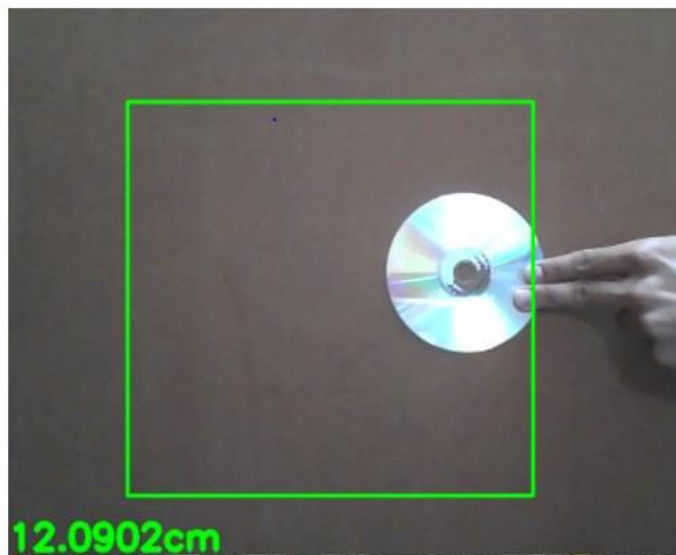


Figura 5: Criando relação pixel/centímetro para iniciar medições.

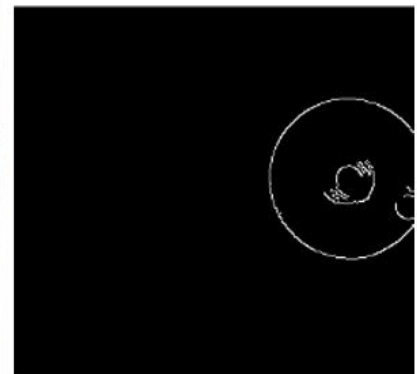


Figura 6: Bordas extraídas de um objeto.

Fala: Na Figura 3, o diâmetro do CD foi equivalente à 134 pixels. Como um CD tem diâmetro de 12,0 centímetros, o software computou a relação pixel/centímetro equivalente à aproximadamente 0,089. Esta referenciação foi realizada a uma distância de 750 milímetros, e mantida para as demais medições.

Fala: Como as bordas que não representam o objeto, e as bordas do objeto são armazenadas da mesma maneira, para realizar as medições era necessário criar um ambiente em que as únicas bordas presentes fossem as do objeto desejado. Essa situação era devido ao fato de o algoritmo procurar o primeiro e o último elemento da matriz que representasse borda, e quando encontrado falsas bordas, ou bordas que não representam o objeto, falhas são geradas. Percebido isso, foi desenvolvido uma nova função no programa que é responsável por delimitar a área de interesse. Esta função tem como simples objetivo reduzir a área de busca por bordas, transformando a matriz da imagem em uma matriz menor em que o objeto se encontra inserido (nas figuras 4 e 5 a região de interesse é a demarcada pelo contorno verde).

NESSA ETAPA SERA EXECUTADO O PROGRAMA PARA DEMONSTRAR OS PASSOS. Inicialmente os primeiros algoritmos foram desenvolvidos utilizando um software chamado SciLab. Essa escolha deveu-se ao fato de esse ser um software que integra linguagem de programação básica, com cálculos matriciais e construção de gráficos. Porém, ficamos limitados num primeiro momento apenas à visualização das formas matriciais que uma imagem recebe ao ser tratada por um compilador. Após compreendido a aplicação da derivada para chegar em uma matriz binária, percebemos que seria possível ir um pouco além, e desenvolver um algoritmo capaz de utilizar essa matriz para computar de forma simples a altura de um objeto. Para essa etapa, ficou decidido à troca do software SciLab por um programa capaz de copilar a linguagem C++. Essa decisão foi tomada pelo fato de essa linguagem trabalhar com a biblioteca OpenCV. E como explicado, essa biblioteca possui eficiência para aplicações em tempo real e, além disso, demanda menos processamento o que melhorou o desempenho do algoritmo. Bom, o algoritmo para medição de objetos não é algo novo no mercado, pois já existem aplicações que trabalham com precisão e tem larga aplicação na robótica, por exemplo. Porém, a decisão de desenvolver esse algoritmo foi para demonstrar que após realizar o processamento de uma imagem, é possível utilizar os dados extraídos para diversas aplicações. Explicando de forma breve têm-se o seguinte: 1. As primeiras funções do programa são, obviamente, focadas na transformação de uma imagem em uma outra contendo apenas bordas. A biblioteca OpenCV já vem com essas funções internas, então apenas as aplicamos (Demonstrar a imagem de bordas em tempo real); 2. É importante ressaltar que a qualquer momento pode-se chamar os comandos para melhorar a

limiarização, pois como veremos nos resultados, impactam bastante na medição (Trabalhar os comandos de limiarização demonstrando a mudança que ocorre nas bordas); 3. Além disso, foi necessário criar uma função para diminuir a área de busca, pois em uma imagem são encontradas não apenas as bordas dos objetos, como também as bordas de elementos externos (Demonstrar redução da área de busca pelo objeto); 4. Na sequência, sabendo que a imagem possui uma matriz binária que a representa, e que os elementos 1 são bordas, e os elementos 0 nulos, criamos uma função que percorre cada elemento da matriz e guarda a primeira e a última linha contendo borda (Utilizar a imagem contendo apenas bordas para ilustrar essa fala); 5. Com essa informação, realizamos a subtração da última linha pela primeira, obtendo a altura do objeto em pixels (Demonstrar a medição de um objeto com a sua altura dada em pixels); 6. Para realizar as medições é necessário criar uma relação inicial entre pixel e centímetros. Para isso utiliza-se um objeto de medições conhecidas para informar ao sistema quanto em centímetros aquela altura em pixels representa (Realizar referência do sistema – Utilizar CD); 7. Após criado essa relação é possível medir os objetos cuja as dimensões se deseja saber (Demonstrar resultados). Limitações: • Por estarmos trabalhando com uma câmera 2D simples, não foi possível trabalhar com profundidade, portanto todo objeto medido deve estar a uma mesma distância do objeto utilizado como referência; • Além disso, ao aplicar rotações as medições se invalidam. (Explicar o porque foi interessante utilizar um CD). Sendo essas limitações possíveis sequencia a serem dadas no projeto. **NESSA ETAPA SERA EXECUTADO O PROGRAMA PARA DEMONSTRAR OS PASSOS.** Inicialmente os primeiros algoritmos foram desenvolvidos utilizando um software chamado SciLab. Essa escolha deveu-se ao fato de esse ser um software que integra linguagem de programação básica, com cálculos matriciais e construção de gráficos. Porém, ficamos limitados num primeiro momento apenas à visualização das formas matriciais que uma imagem recebe ao ser tratada por um compilador. Após compreendido a aplicação da derivada para chegar em uma matriz binária, percebemos que seria possível ir um pouco além, e desenvolver um algoritmo capaz de utilizar essa matriz para computar de forma simples a altura de um objeto. Para essa etapa, ficou decidido à troca do software SciLab por um programa capaz de copilar a linguagem C++. Essa decisão foi tomada pelo fato de essa linguagem trabalhar com a biblioteca OpenCV. E como explicado, essa biblioteca possui eficiência para aplicações em tempo real e, além disso, demanda menos processamento o que melhorou o desempenho do algoritmo. Bom, o algoritmo para medição de objetos não é algo novo no mercado, pois já existem aplicações que trabalham

com precisão e tem larga aplicação na robótica, por exemplo. Porém, a decisão de desenvolver esse algoritmo foi para demonstrar que após realizar o processamento de uma imagem, é possível utilizar os dados extraídos para diversas aplicações. Explicando de forma breve têm-se o seguinte: 1. As primeiras funções do programa são, obviamente, focadas na transformação de uma imagem em uma outra contendo apenas bordas. A biblioteca OpenCV já vem com essas funções internas, então apenas as aplicamos (Demonstrar a imagem de bordas em tempo real); 2. É importante ressaltar que a qualquer momento pode-se chamar os comandos para melhorar a limiarização, pois como veremos nos resultados, impactam bastante na medição (Trabalhar os comandos de limiarização demonstrando a mudança que ocorre nas bordas); 3. Além disso, foi necessário criar uma função para diminuir a área de busca, pois em uma imagem são encontradas não apenas as bordas dos objetos, como também as bordas de elementos externos (Demonstrar redução da área de busca pelo objeto); 4. Na sequência, sabendo que a imagem possui uma matriz binária que a representa, e que os elementos 1 são bordas, e os elementos 0 nulos, criamos uma função que percorre cada elemento da matriz e guarda a primeira e a última linha contendo borda (Utilizar a imagem contendo apenas bordas para ilustrar essa fala); 5. Com essa informação, realizamos a subtração da última linha pela primeira, obtendo a altura do objeto em pixels (Demonstrar a medição de um objeto com a sua altura dada em pixels); 6. Para realizar as medições é necessário criar uma relação inicial entre pixel e centímetros. Para isso utiliza-se um objeto de medições conhecidas para informar ao sistema quanto em centímetros aquela altura em pixels representa (Realizar referência do sistema – Utilizar CD); 7. Após criado essa relação é possível medir os objetos cuja as dimensões se deseja saber (Demonstrar resultados). Limitações: • Por estarmos trabalhando com uma câmera 2D simples, não foi possível trabalhar com profundidade, portanto todo objeto medido deve estar a uma mesma distância do objeto utilizado como referência; • Além disso, ao aplicar rotações as medições se invalidam. (Explicar o porque foi interessante utilizar um CD). Sendo essas limitações possíveis sequencia a serem dadas no projeto.